INTRODUCTION

The Gemini GM851 12-bit A/D board provides fast analogue to digital conversion on 80-BUS based microcomputer systems. Its features include:-

- Selectable choice of 16 single-ended or 8 differential inputs

- Fast and accurate sample/hold amplifier to ensure conversion integrity.

- High-accuracy instrumentation amplifier to provide selectable gains of 1, 10, 100 and 1000.

- Selectable choice of input ranges for the A/D of 0V to 10V, 0V to 20V, -5V to +5V and -10V to +10V. In addition, a further set is available, namely 0V to 10.24V, 0V to 20.48V, -5.12V to +5.12V and -10.24V to +10.24V. This latter set may be more convenient as it yields conversion increments of 5 mV/bit and 2.5 mV/bit.

- The choice of input voltage ranges in conjunction with the gain selection provided by the instrumentation amplifier gives a highly versatile analogue input range.

- Selectable choice of full (12-bit) resolution conversion, or a shortened conversion of 8-bits

- Conversion time in the order of 30 uSecs for a 12-bit conversion.

- The I/O address of the board is user configurable and an on-board adjust/trim for zero and gain is provided.

- Full flag/interrupt control is provided enabling multiple independent tasks to use the board.

- Physical details: The board measures 8" x 8" and is 80-BUS. It is a quality double-sided PCB with silk screening, solder resist and a gold-plated edge connector.

Guarantee

Your GM851 A/D Board is guaranteed by the supplier (your dealer) for one year from the date of purchase. This warrants that the hardware when dispatched will be free from defects in materials and workmanship. No liability is assumed for damages whether consequential damages or loss of profits or otherwise arising out of or in connection with the use or performance of the hardware. Any faults attributable to the incorrect installation of the board within your system will be fully chargeable as to both parts and labour. This guarantee extends as far as the original hardware as supplied and no work on GM851 A/D boards modified in any way will be carried out.

Any queries regarding the installation and operation of your GM851 A/D board should be directed at your Gemini dealer

THE A/D CONVERTER AND SAMPLE AND HOLD TECHNIQUES

The A/D converter used in the circuit uses the technique of successive approximation in order to convert the analogue input signal into its digital representation.

Successive approximation is a conversion technique where in an 8-bit converter, for example, bit 7 (the most significant bit) has a 'weighting' of half the input range of the device. In this case - half of 5V, or 2.5V. Bit 6 has a weighting of half again (1.25V), and so on. To convert an input, bit 7 is set (on) and the resulting voltage (2.5V) is compared to the input. If the input voltage is greater, the bit is left set; and if less, then it is cleared. Next bit 6 is set and the resulting voltage, which is the sum of 1.25 plus the previous state of bit 7 (i.e. 2.5V or 0V), is compared to the input. If the input voltage is higher, but 6 is left set; if lower, it is cleared.

This procedure is continued for all the bits in the converter. On completion, the states of the bits give the digital representation of the analogue input voltage.

It is important that the input voltage being sampled and converted using this technique does not vary by an amount greater than the resolution of the A/D device being used whilst the conversion is taking place. This would render the conversion inaccurate.

In an 8-bit converter with an input range of 0-5 Volts, the input signal should not vary by more than 9.8 mV during conversion. (A voltage equal to the weighting of half the least significant bit, bit 0.) Most input signals will in fact, vary during conversion, and it is for this reason that a circuit is employed which will take a sample of the input voltage of a given input channel (selected by the multiplexer) and hold it, - a sample and hold circuit.

This circuit now presents a stable sample of the input voltage to the A/D converter during its conversion time, even though the input voltage at the selected input to the board may be varying.

A/D boards which do not have a sample and hold circuit will not convert varying input signals accurately if the fluctuation of the input signal is such that the voltage of that signal varies by an amount greater than 9.8 mV in the above example. Calculations show that signals with a frequency greater than 10 Hz will cause significant errors in A/D boards with no sample and hold circuit. With 12-bit A/D boards, such as the GM851, the sample and hold circuitry is obviously of even greater importance.

INSTALLATION

The GM851 A/D board is 80-BUS compatible, i.e. the board may be plugged straight into the BUS and no extra connections are required to or from the board apart from the analogue input signals. However, one or two points must be noted and checked before attempting to use the board.

A/D Board Address

Communication to and from the A/D board is performed using an on board PIO. The port addresses have been pre-selected to lie in the range 30H - 33H. However, these may re re-configured by the user to suit the needs of his particular system.

The switch block, SW1 (alongside IC 10), selects the I/O block address of the board. Appendix 1 shows the full address mapping that is possible.

Clock Frequency

The 80-BUS CPU clock is assumed to be 4MHz, and this will normally be the case. If another clock rate is in use then LK1 and LK2 need to be set accordingly. Refer to the circuit diagrams and/or your Gemini dealer for further details.

INPUT CONNECTOR

A 34-way right-angle ID connector is supplied for connecting inputs to the GM851. The connections to this are given below. Note that the Pin 1 end of the connector is marked by a small triangular arrow head on the connector. The "odd" connections are on the top row, and the "even" connections are on the bottom row, adjacent to the PCB.

Single ended input	Differential input		Connector pins		
9	1B		1	2	:
1	1A		3	4	:
10	2B		5	6	:
2	2A		7	8	:
11	3B		9	10	:
3	3A		11	12	:
12	4B		13	14	:
4	4A		15	16	:- Ground (0V)
13	5B		17	18	:
5	5A		19	20	:
14	6B		21	22	:
6	6A		23	24	:
15	7B		25	26	:
7	7A		27	28	:
16	8B		29	30	:
8	8A		31	32	:
		+15V	33	34	-15V

SETTING THE SWITCH AND LINK OPTIONS

The GM851 is fitted with a number of links and switches to set various board options The following sections describe the use of these.

Input mode

The GM851 can be set to cater for 16 single-ended or 8 differential inputs. The differential option should always be used when measuring low level input signals in a noisy environment as it provides far higher noise immunity. The "Input Connector" section above gives the input pins for both the differential and single-ended options.

The input mode is controlled by switches 5, 6 and 7 on the switch block adjacent to the PCB edge and LK4, adjacent to IC3.

To select 16 single-ended inputs then:

switch 5 (A)	= ON
switch 6 (B)	= OFF
switch 7 (LK3)	= ON
LK4	= link inserted to connect pin 2 to pin 3

To select 8 differential inputs then:

switch 5 (A)	= OFF
switch 6 (B)	= ON
switch 7 (LK3)	= OFF
LK4	= link inserted to connect pin 1 to pin 2

On-board amplifier

The GM851 is supplied with a high precision amplifier on board to amplify the input signals before they reach the A/D convertor. The input impedance of the amplifier is 10 to the power 9 ohms.

The gain of the amplifier is controlled by the 8-way switch block on the edge of the GM851 board. Switches 1 to 4 are used to select the relevant gain. There are 4 "pre-set" gains of x1, x10, x100 and x1000. In addition a user-specific gain can be produced by inserting a resistor into the board at the position alongside the switches and marked R8. The required resistor value is:

R8 value = 40000/(GAIN-1)

NOTES: 1) This option should be used with caution because of possible mismatch between the external R and internal R. Its use is not recommended unless unavoidable.

2) Due to an error on GM851 Issue 1 PCBs, switch 1 must always be OFF. With Issue 1 boards the R8 option is enabled by switching OFF the other switches and inserting the resistor; for all other gain options R8 must NOT be fitted.

For the different gain options the switches should be set as follows:

Gain	1	2	3	4
x1	OFF	OFF	OFF	OFF
x10	OFF	ON	OFF	OFF
x100	OFF	OFF	ON	OFF
x1000	OFF	OFF	OFF	ON
Using R8	ON	OFF	OFF	OFF

With the x1, x10 and x100 settings the settling time of the amplifier is typically 15 uSec to 0.01% for a 20V swing. The x1000 setting gives a settling time of 75 uSec for a 20V swing, whereas the GM851 design only allows 16 uSec. Caution is advised in using the x1000 gain with large voltage swings.

A/D Input Range and Polarity

Input Range

The overall input voltage range (i.e. total voltage swing) to the A/D convertor (AFTER any amplification) may be set by switches 1 to 4 on LK6:

	Range	(Polarity options)	1	2	3	4
(a)	10V	(-5V to +5V or 0V to +10V)	ON	OFF	OFF	OFF
(b)	10.24V	(-5.24V to +5.24V or 0V to +10.24V)	OFF	ON	OFF	OFF
(c)	20V	(-10V to +10V only)	OFF	OFF	ON	OFF
(d)	20.48V	(-10.24V to +10.24V only)	OFF	OFF	OFF	ON

The 10.24V and 20.48V ranges are provided as they give a "round" number of mV per conversion bit. e.g. 12-bit = 4096 levels. 20.48V/4096 = 5 mV per bit.

Polarity

Note that the input range may be selected to be unipolar or bipolar. This is selected by LK7 and LK8, which are adjacent to the switch block LK6. The input range selected by switch block LK6 is modified accordingly.

Unipolar - LK7 absent, LK8 present

Input ranges become: (a) 0V to +10V, (b) 0V to +10.24V.

NOTE: The (c) and (d) ranges are ILLEGAL unipolar options.

Bipolar - LK7 present, LK8 absent

Input ranges become: (a) -5V to +5V, (b) -5.12V to +5.12V,
(c) -10V to +10V, (d) -10.24V to +10.24V

OPERATING THE BOARD - SOFTWARE CONTROL

Simple Mode

The A/D board plugs directly into the BUS and communication to and from the board is via an on board Z80A PIO using its twin 8 bit parallel ports.

These ports are referred to as "A" and "B" and both have control and data registers. Both of these ports are I/O mapped to the system and may be accessed by OUTputting and INputting from the host processor.

To modify the port address refer to the installation section.

The preset I/O mapping is as follows:

Address-(Hex)	Port
30h	"A" DATA
31h	"B" DATA
32h	"A" CONTROL
33h	"B" CONTROL

For a detailed description of the control and usage of a Z80A PIO, refer to a Zilog/Mostek PIO manual. Such a detailed knowledge is not required to operate the A/D board as all basic PIO programming information is given in this document, but reading the manual will explain the exact state that the PIO has been put into. For example, you may need to know more details if you intend to use the spare PORT B lines provided by the PIO.

The following program will initialise the A/D board and demonstrate channel sampling in simple mode. A detailed description follows the listing.

```
         LD A,4FH
         OUT (32H),A          ;PORT A - INPUT MODE 1
         LD A,0CFH
         OUT (33H),A          ;PORT B - CONTROL MODE 3
         LD A,0C0H
         OUT (33H),A          ;I/O BYTE
         IN A,(30H)           ;DUMMY READ
         LD A,06H             ;TYPE I CONVERSION ON ANALOGUE INPUT 7
         OUT (31H),A          ;OUTPUT TO B DATA, SETS A/D BOARD TO TYPE I AND
                              ;STARTS CONVERSION OF ANALOGUE INPUT 7

    L1   IN A,(31H)           ;READ B DATA
         AND 80H              ;MASK TO TEST EOC BIT
         JR Z,L1              ;LOOP UNTIL SET
         in a, (30H)          ;READ RESULT - least significant bits in 7-4
         ld c,a               ; move to c (for example)
         in a, (30H)          ; read most significant bits in 7-0
                              ;THIS STARTS NEXT CONVERSION CYCLE  (TYPE I)
         USER ROUTINE         ;eg, display result, etc
         USER ROUTINE
         USER ROUTINE
         JP L1                ;LOOP FOR ANOTHER RESULT
```

Setting up the PIO

Place PORT A into INPUT MODE 1 by outputting 4Fh or 7Fh to PORT A CONTROL.

```
LD A,4FH
OUT (32H),A
```

Place PORT B into CONTROL MODE 3 by outputting CFh or FFh to PORT B CONTROL.

```
LD A,0CFH
OUT (33H),A
```

This output must then be followed by an I/O byte defining which bits of PORT B DATA will be used for input, and which for output. In most cases this will be bits 0-5 for output and bits 6-7 for input, ie, 11000000 (0 = output, 1 = input).

The I/O byte is loaded into the PIO as follows:

```
LD A,0C0H
OUT (33H),A
```

Interpreting the I/O byte

```
B7     B6     B5     B4     B3     B2     B1     B0
1      1      0      0      0      0      0      0
¦      ¦      ¦      ¦      ¦      ¦      ¦      ¦
¦      ¦      ¦      ¦      ¦      ¦      ¦      ¦
EOC    ¦      ¦      ¦      --------------------------
    READ      ¦     not                channel address
              ¦     used
              ¦
            TYPE
```

(EOC - End of Conversion)

Assuming the byte was set up as above (6 outputs - bits 0-5, 2 inputs bits 6-7), then the outputs may be used by outputting a value to PORT B DATA. Outputting an 8-bit byte to this port will not affect those bits defined as inputs.

Bits 0-3 are used to select the CHANNEL ADDRESS for the on board multiplexer, and therefore the input channel to be used. For example, if the bits B3 - B0 are all set to 1, analogue input number 16 will be selected. If the bits B3 - B0 are all cleared, then analogue input number 1 will be selected, and so on.

Bit B4 is unused.

Bit B5 - Type:- this bit is used to select one of two methods of initiating a conversion cycle, namely:

Type I (B5=0). Second read of PORT A DATA or write PORT B DATA will initiate an A/D conversion.

Type II (B5=1). Write PORT B DATA and ONLY write PORT B DATA will initiate an A/D conversion.

A Type I conversion can be used where many samples of a selected input channel are to be taken. The cycle can be started by writing the required CHANNEL ADDRESS to PORT B DATA, (ensuring that bit 6 remains cleared). Further samples and conversions of the selected channel will be started automatically upon reading the complete result of the previous conversion (i.e. both MSB and LSB), - the result being held in PORT A DATA.

A Type II conversion will only be initiated by writing the required CHANNEL ADDRESS to PORT B DATA (ensuring that bit 6 remains set).

When PORT B DATA is read, the EOC and READ bis (B6-B7) are interpreted together as shown in the following table:

EOC	READ	
0	0	- conversion in progress
0	1	- NOT ALLOWED (will not occur)
1	0	- conversion finished,- DATA has been read (RESET STATE)
1	1	- conversion finished,- DATA has not been read

The significance of the EOC flag is that it signals that the requested conversion has been finished. The significance of the data has/has not been read flag is that a program may use this flag to check if other routines which are using the A/D board have read the result of the last conversion they requested. This extra flag permits many programs (tasks) to use the A/D board without destroying the other task's data.

The final step of setting up the PIO is to issue a dummy read instruction. This ensures that the PIO line ARDY is in the correct state required by the A/D board. This need only be done once after setting up the PIO and is done as follows:

```
IN A,(30H)
```

The explanation of the requirement of a dummy read is given in the Zilog/Mostek manual.

Initiating an A/D conversion cycle

For this example we will assume analogue input 7 is to be sampled (J7, CHANNEL ADDRESS 06H) and that a Type I conversion is to be initiated.

This is done as follows:

```
LD A 06H            ;Type I conversion on analogue input 7
OUT (31H) A         ;output  to B DATA - this sets A/D board to Type I
                    ;and starts conversion of analogue input 7
```

Waiting for end of conversion

The end of conversion is signified by bit 7 of the PORT B DATA being set as described earlier and may be checked for as follows:

```
L1    IN A (31H)          ;read B DATA
      AND 80H             ;mask off all except top bit (EOC bit)
      JR Z L1             ;loop until EOC set, ie, conversion finished
```

When the EOC bit becomes set, then this signifies that the A/D conversion on analogue channel 7 has finished and a value may then be extracted by reading PORT A data. This is the digital result of the first conversion.

```
in a, (30H)         ; read result - least significant bits in 7-4
ld c,a              ; move to c, for example
in a, (30H)         ; most significant bits in bits 7-0
                    ; this starts next conversion (Type 1)
USER ROUTINE        ;e.g. display result or store it
JP L1               ;loop to look for another result
```

If a Type II conversion had been selected, then the B DATA would have to be written to each time a conversion was required.

Interrupt Mode

Operating the A/D board in its simple mode will be adequate for the majority of applications where the host compuer is dedicated to the sampling of data. It can be seen, however, from the previous example, that a portion of processor time is spend idling in a loop waiting for a conversion to complete. There is a more efficient way of gathering data, - by using interrupts.

For a more detailed description of how your processor and PIO handles interrupts, consult your technical manuals once again, - but the concept is quite simple.

It is possible to program the PIO such that when a signal is received from the outside world, the processor is interrupted from its current task, and control is passed to a pre-determined address. This address is the start address of a user-written program called the "interrupt service routine". This routine may then take some action as a result of the incoming signal, - and may then return to its previous activity having "serviced the interrupt".

The A/D board may be set up to interrupt the host processor when it has some data for it.

To employ interrupt control on the A/D board, one would have to set up the PIO in a fashion not too dissimilar from before, but in this case setting up ready for interrupt generation, using the MODE 2 interrupt response.

The host processor may then continue in some other activity, such as controlling a piece of equipment, and when an interrupt is received from the A/D board, - acts on the data received from the conversion, eg, by modifying its control program, and then returning to its previous task ready to accept further interrupts.

Example:

```
          LD A,4FH
          OUT (32H),A          ;PORT A - INPUT MODE 1
          LD A 0CFH
          OUT (33H),A          ;PORT B - CONTROL MODE 3
          LD A,0C0H
          OUT (33H),A          ;I/O BYTE
          LD A,0CH             ;HIGH ORDER ADDRESS BYTE OF -
          LD I,A               ;INTERRUPT VECTOR
          LD A,0B7H
          OUT (33H),A          ;INTERRUPT CONTROL WORD
          LD A,7FH
          OUT (33H),A          ;INTERRUPT MASK
          IM 2                 ;INTERRUPT MODE 2
          IN A,(30H)           ;DUMMY READ
          EI                   ;ENABLE INTERRUPTS
          LD A,(06H)           ;TYPE I CONVERSION ON ANALOGUE INPUT 7
          OUT (31H),A          ;OUTPUT TO B DATA, SETS A/D BOARD TO TYPE I AND
                               ;STARTS CONVERSION OF ANALOGUE INPUT 7
     L1   USER ROUTINE
          USER ROUTINE
          JP L1                ;LOOP AWAITING INTERRUPTS
```

```
ISR  IN,A (30H)          ;INTERRUPT SERVICE ROUTINE
                         ;READ RESULT FROM CONVERSION
     CALL DISP           ;DISPLAY RESULT, ET
     EI                  ;RE-ENABLE INTERRUPTS
     RETI                ;RETURN TO MAIN PROGRAM
```

Setting up the PIO

```
     LD A,4FH            ;PORT A into INPUT MODE 1
     OUT (32H),A         ;PORT A CONTROL

     LD A,0CFH           ;PORT B into CONTROL MODE 3
     OUT (33H),A         ;PORT B CONTROL
```

This output is then followed by the PORT B I/O byte as before (ie, bits 0-5 output, bits 6-7 input).

```
     LD A,0C0H
     OUT (33H),A
```

MODE 2 Interrupts

A detailed description of the mode 2 interupt response is given in the MOSTEK/ZILOG manuals. In mode 2, the CPU will respond to an interrupt by jumping to an interrupt service routine whose address is stored in an area of memory selected by the programmer to contain the address(es) of the interrupt service routine(s), known as a table of addresses.

This table is used when there are one or more peripheral devices (in this case, the A/D board), interrupting the Z80, and therefore one or more interrupt service routines (ISR). The table consists of a list of 16 bit addresses of all the interrupt service routines in the form:

```
     low  order byte of the address of ISR 1
     high order byte of the address of ISR 1
     low  order byte of the address of ISR 2
     high order byte of the address of ISR 2
     etc, etc
```

This table may be changed at any time by the programmer.

When an interrupt occurs, the CPU forms a 16 bit pointer to obtain the desired interrupt service routine starting address from within the table. The high order part of this pointer is obtained from the "I" register contents. The I register must have previously been set up by the programmer, (eg, LD I A). (Note that the table must therefore be less than 256 addresses long as the high order part of the pointer must remain constant, ie, "I" is only set up once.)

The low order portion of the pointer DOES change, of course, to give the final address of the table entry. This byte is supplied by the interrupting device. In the case of the A/D board, it must be passed to the A/D PIO as is shown in the example.

Note that the Z80 PIO includes a daisy chain priority interrupt structure that automatically supplies the programmed low order vector to the CPU during interrupt acknowledge. Refer to the Z80 PIO manual for details.

Note that all addresses (ie, the address of each two-byte pair within the table, and the addresses of the ISR's themselves) must all start on two-byte boundaries (even addresses). Thus their least significant bits should be zero.

In the example given here, the address within the table which contains the address of the interrupt service routine is 0CD0H, so we must load the lower 8 bits of the address into the PIO on the A/D board. This is done as follows:

```
LD A,0D0H            ;low order 8-bits of table address
OUT (33H),A          ;STORE IN PIO PORT B INTERRUPT VECTOR
```

Now the high order address byte of the interrupt vector is stored in the I register:

```
LD A,0CH             ;high order byte = 0CH
LD I,A
```

Setting up the Interrupt Control Word

Having set up the interrupt vector, an interrupt control word is output to PORT B CONTROL:

```
LD A 0B7H            ;INTERRUPT CONTROL WORD
OUT (33H) A          ;PORT B CONTROL
```

This control word is made up as follows:

```
     B7   B6   B5   B4   B3   B2   B1   B0
     1    0    1    1    0    1    1    1
     ¦    ¦    ¦    ¦    ¦    ¦    ¦    ¦
     ¦    ¦    ¦    ¦    ¦    ¦    ¦    ¦
     ¦    ¦    ¦    ¦    ----------------
  enable  ¦    ¦    ¦    signifies interrupt control word
  inter-  ¦    ¦    ¦
  rupts   ¦    ¦    indicates mask to follow
          ¦    ¦
          ¦    high level/low level
          ¦
          "OR" or "AND"
```

Bit 7 enables interrupts from the PIO when set (1), and disables them when cleared (0).

Bit B6 (OR or AND) defines the logical operation to be performed in port monitoring. If B6 = 1, an AND function is specified and if B6 = 0, an OR function is specified.

For example, if the AND function is specified, ALL non-masked bits (see later) corresponding to input lines to the port must go to the specified state (selected by B5) before an interrupt will be generated.

The OR function will allow an interrupt to be generated if any non-masked bits go to the selected active state.

Bit B5 denotes whether the interrupt signal is to be generated by a transition of the input line selected from:

high to low (0) or
low to high (1)

ie, it defines the "active state" of the input lines. In this example a transition from low to high is selected.

The interrupt mask now follows:

```
LD A 7FH            ;interrupt mask
OUT (33H) A         ;PORT B CONTROL
```

This mask is made up as follows:

B7	B6	B5	B4	B3	B2	B1	B0
0	1	1	1	1	1	1	1

Only those port lines whose mask bit is zero will be monitored for generating an input.

Bits 6 and 7 of the mask signify the following when bit 6 of the control word is "0", ie, an OR function selected, or "1", ie, an AND function selected.

B7	B6		
0	1	-	generate interrupt on end of conversion
1	0	-	generate interrupt on read

When the PIO is set up as in this example, a transition of the PORT B input line B7 from low to high (signifying the end of a conversion) will cause an interrupt to be generated. Should bit 7 of the mask be set and bit 6 cleared and bit 5 cleared (ie, looking for a transition from high to low), then an interrupt would be generated when the data had been read.

Should bit 7 and bit 6 both be cleared then an interrupt will occur dependent upon the AND/OR conditions selected by bit 6 of the control word and the active state selected. If both bit 7 and bit 6 are set then all the inputs are masked and an interrupt will not occur.

Continuing the example:

The CPU is now set to interrupt mode 2

```
IM 2
```

Now a dummy read to ensure ARDY is in the correct state.

```
IN A,(30H)
```

The "setting up" is now complete and the user program begins.

```
     EI                  ;enable interrupts
     LD A,06H            ;TYPE I CONVERSION ON ANALOGUE INPUT 7
     OUT (31H),A         ;OUTPUT TO B DATA, SETS A/D BOARD TO TYPE I AND
                         ;STARTS CONVERSION OF ANALOGUE INPUT 7

L1   user prog
     .....
     .....
     .....
     .....
     JP L1               ;loop round user prog waiting for interrupts
                         ;this could be a display routine or a control prog
                         ;using values from latest conversion
```

Now the code of the Interrupt Service Routine

```
ISR  IN A (30H)          ;interrupt service routine
                         ;read result from conversion
                         ;this starts another conversion (Type I)
     CALL DISP           ;put result into a variable ..
                         ;or perhaps just display it

     EI                  ;re-enable interrupts
     RETI                ;return to main program
```

ISR Start-Address Table

Assuming no other interrupt driven devices in the system, the table has only one entry:-

OCD0/1 - contains the 16 bit address of the ISR start.

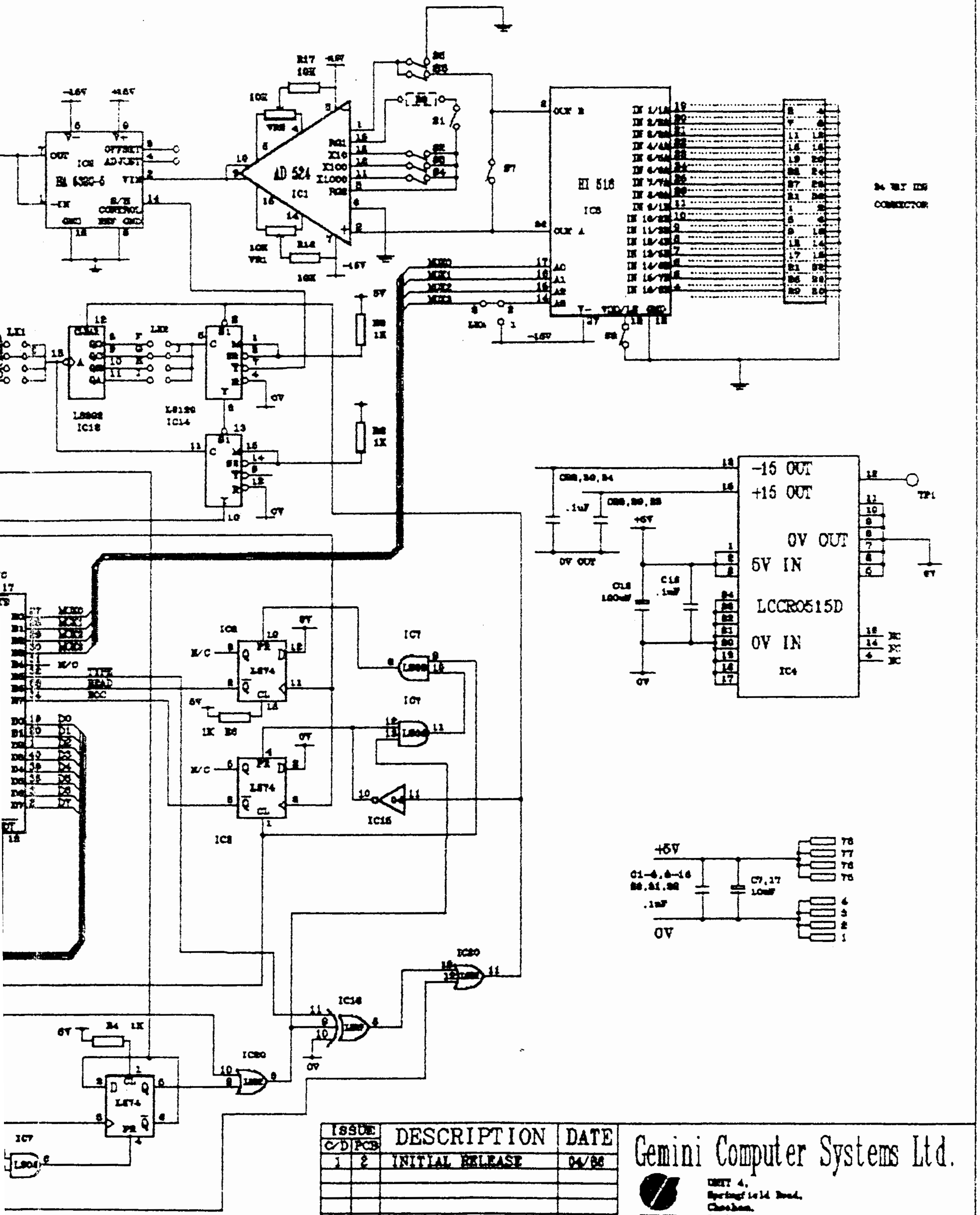
Address mapping

	Address		Switch	Address		Switch
			SW1			SW1
	DEC	HEX	654321	DEC	HEX	654321
	0-3	00-03	000000	128-131	80-83	100000
	4-7	04-07	000001	132-135	84-87	100001
	8-11	08-0B	000010	136-139	88-8B	100010
	12-15	0C-0F	000011	140-143	8C-8F	100011
	16-19	10-13	000100	144-147	90-93	100100
	20-23	14-17	000101	148-151	94-97	100101
	24-27	18-1B	000110	152-155	98-9B	100110
	28-31	1C-1F	000111	156-159	9C-9F	100111
	32-35	20-23	001000	160-163	A0-A3	101000
	36-39	24-27	001001	164-167	A4-A7	101001
	40-43	28-2B	001010	168-171	A8-AB	101010
	44-47	2C-2F	001011	172-175	AC-AF	101011
+	48-51	30-33	001100	176-179	B0-B3	101100
	52-55	34-37	001101	180-183	B4-B7	101101
	56-59	38-3B	001110	184-187	B8-BB	101110
	60-63	3C-3F	001111	188-191	BC-BF	101111
	64-67	40-43	010000	192-195	C0-C3	110000
	68-71	44-47	010001	196-199	C4-C7	110001
	72-75	48-4B	010010	200-203	C8-CB	110010
	76-79	4C-4F	010011	204-207	CC-CF	110011
	80-83	50-53	010100	208-211	D0-D3	110100
	84-87	54-57	010101	212-215	D4-D7	110101
	88-91	58-5B	010110	216-219	D8-DB	110110
	92-95	5C-5F	010111	220-223	DC-DF	110111
	96-99	60-63	011000	224-227	E0-E3	111000
	100-103	64-67	011001	228-231	E4-E7	111001
	104-107	68-6B	011010	232-235	E8-EB	111010
	108-111	6C-6F	011011	236-239	EC-EF	111011
	112-115	70-73	011100	240-243	F0-F3	111100
	116-119	74-77	011101	244-247	F4-F7	111101
	120-123	78-7B	011110	248-251	F8-FB	111110
	124-127	7C-7F	011111	252-255	FC-FF	111111

1 = Switch ON

0 = Switch OFF

+ = Default setting of GM851 board as supplied



ISSUE C/D	ISSUE PCB	DESCRIPTION	DATE
1	2	INITIAL RELEASE	04/86

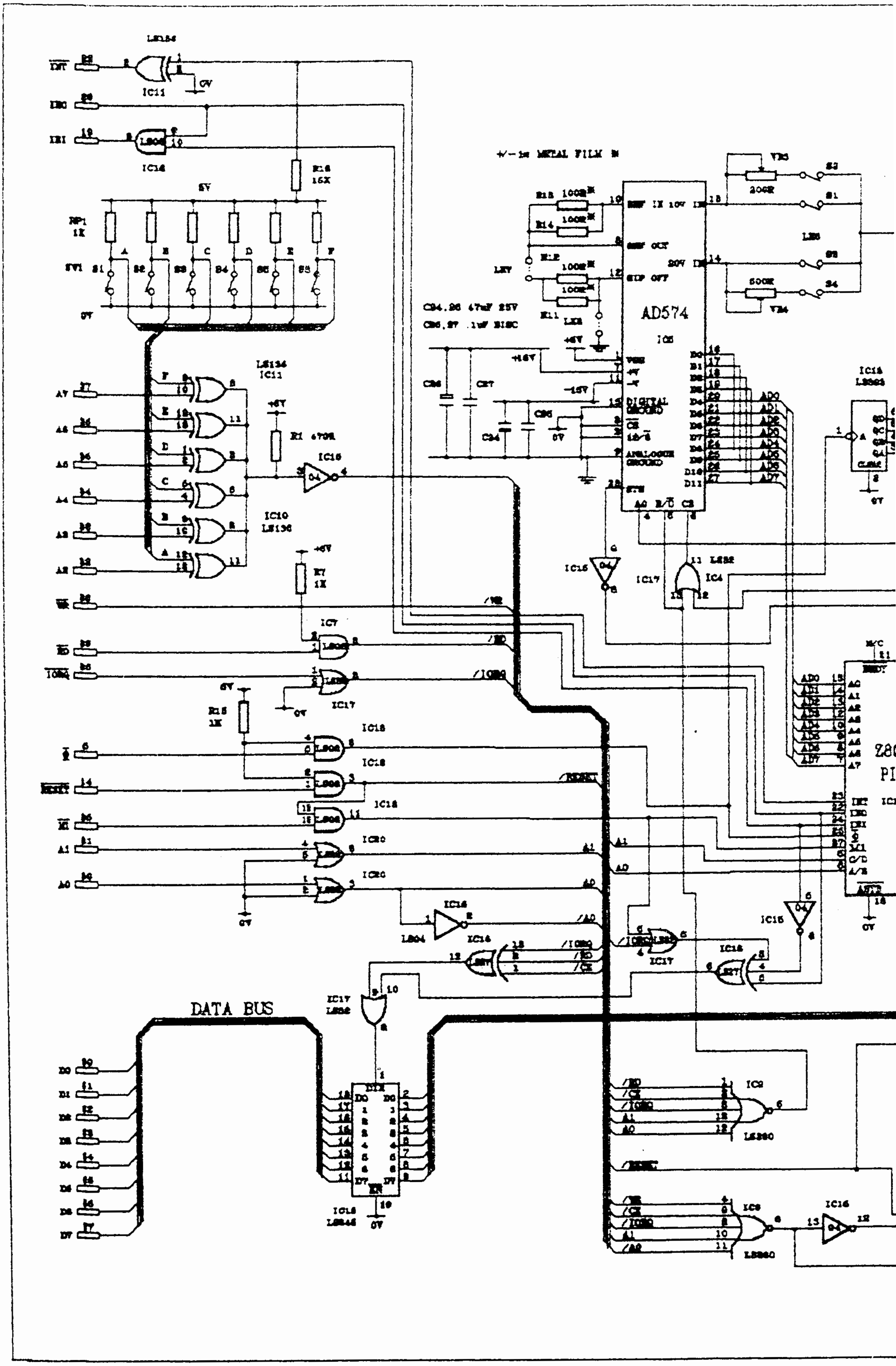
Gemini Computer Systems Ltd.

UNIT 4,
Springfield Road,
Chesham,
Bucks.
HP5 1PW
(0494) 791010

DRAWN	CHK'D	APPV'D	TRC'D	DATE
MA	PGM			

TITLE GM 861

12 BIT ANALOGUE TO DIGITAL CONVERTER


INT
IEO
IEI
LS136
IC11
0V
LS08
IC16
R16
15K
5V
RP1
1K
SW1 S1
S2
S3
S4
S5
S6
A
B
C
D
E
F
0V
LS136
IC11
A7
A6
A5
A4
A3
A2
+5V
R1 470R
IC15
04
IC10
LS136
+5V
R7
1K
WR
RD
IORQ
IC7
LS08
LS32
IC17
0V
5V
R15
1K
IC18
LS08
Φ
RESET
M1
A1
A0
IC20
LS32
0V
IC16
LS04
IC14
IC17
LS32
/WR
/RD
/IORQ
/RESET
/A0
/CE
DATA BUS
D0
D1
D2
D3
D4
D5
D6
D7
DIR
EN
IC19
LS245
0V
+/- 1% METAL FILM R
R13 100R
R14 100R
R12 100R
R11
LK1
LK2
+5V
REF IN 10V IN
REF OUT
20V IN
BIP OFF
AD574
IC5
VR3
20K
VR4
500K
S1
S2
S3
S4
LK3
C24, 25 47uF 25V
C26, 27 .1uF DISC
+15V
-15V
C26
C27
C24
C25
VCC
+5V
DIGITAL GROUND
CE
12/8
ANALOGUE GROUND
STS
A0 R/C CS
DB0
ADO
AD1
AD2
AD3
AD4
AD5
AD6
AD7
D11
IC15
LS32
IC4
IC17
IC13
LS393
CLEAR
0V
M/C
RDY
A0
A1
A2
A3
A4
A5
A6
A7
Z80
PIO
INT
IEO
IEI
M1
C/D
A/B
ASTB
0V
IC9
LS260
/RD
/CE
/IORQ
A1
A0
/RESET
/WR
/A0
IC16
04
LS260